ĐỒ ÁN CUỐI KỲ  
CNMTPTPM

GV: Ngô Ngọc Đăng Khoa

# OVERVIEW

1. Sinh viên thực hiện đồ án theo nhóm
2. Yêu cầu commit lên GITHUB đầy đủ
3. Thời gian vấn đáp dự kiến: theo lịch thi cuối kỳ của trường

**Yêu cầu:** xây dựng hệ thống đặt xe máy qua tổng đài điện thoại ở mức độ cơ bản, gồm nhiều phân hệ như mô tả bên dưới.

# TECHNIQUES

1. SOA RESTful API
2. Realtime Web Application
3. Google Maps API

# APP #1: REQUEST RECEIVER

**WebApp** nhận thông tin khách (request): họ tên, điện thoại, địa chỉ đón khách, ghi chú

1. Vì áp lực số lượng khách, nhân viên có thể gõ địa chỉ rất vắn tắt chứ không phải là 1 địa chỉ “chuẩn”, địa chỉ này là đủ để con người đọc hiểu. VD: 227 nvcu q5 / dhkhtn.

# APP #2: LOCATION IDENTIFIER

**WebApp** xác định toạ độ khách dựa trên thông tin do app #1 ghi nhận

1. Lần lượt thể hiện thông tin các request được ghi nhận bởi app #1 => nhân viên xác định 1 cách tương đối vị trí khách trên bản đồ (dựa vào địa chỉ & ghi chú)
2. Nhân viên sử dụng **geocoding** để xác định nhanh toạ độ dựa vào địa chỉ
3. Nhân viên có thể di chuyển vị trí khách trên bản đồ 1 cách tự do, địa chỉ khách khi đó phải được cập nhật lại tương ứng (**reverse geocoding**)
4. Lưu ý: địa chỉ gửi cho xe là địa chỉ gốc được ghi nhận bởi điện thoại viên (app #1), không phải địa chỉ có được sau khi **reverse geocode**

# APP #3: REQUEST MANAGEMENT

**WebApp** thể hiện danh sách request cùng trạng thái tương ứng (chưa được định vị, đã định vị xong, đã có xe nhận, đang di chuyển, đã hoàn thành, …)

1. Danh sách được sắp xếp theo thứ tự giảm dần theo thời điểm đặt
2. Trong trường hợp request đã có xe nhận, nhân viên có thể chọn xem đường đi ngắn nhất từ xe đến khách trên bản đồ, thông tin tài xế cũng được thể hiện đầy đủ trên danh sách

# APP #4: DRIVER

**WebApp**

1. Giao diện được thiết kế phù hợp với **màn hình điện thoại**
2. Cho phép cập nhật vị trí hiện tại thông qua hành vi click bản đồ. Nếu click xa quá 100m (theo công thức Harversine) thì thông báo lỗi
3. Cho phép tài xế đăng nhập vào hệ thống và sẵn sàng nhận thông tin request
4. Cho phép đổi trạng thái ONLINE / OFFLINE
5. Khi thông tin 1 request được gửi xuống, app thể hiện trong vòng 10s và yêu cầu tài xế phản hồi. Nếu tài xế **TỪ CHỐI** hoặc **KHÔNG PHẢN HỒI**, hệ thống tự động tìm xe khác cho khách.
   1. Khi tài xế đồng ý đón khách, app thể hiện đường đi ngắn nhất từ vị trí hiện tại đến vị trí khách trên bản đồ, giúp tài xế dễ dàng đi đến ĐIỂM ĐÓN KHÁCH.
6. Tài xế khi đón được khách, chọn lệnh BẮT ĐẦU; và sau khi đến nơi, chọn lệnh KẾT THÚC
   1. Sau khi request kết thúc, tài xế được chuyển về trạng thái ONLINE để có thể nhận được request khác

# NOTES

1. Hệ thống chỉ tìm xe ONLINE để gửi request, không tìm xe BUSY (đang đi đón khách, đang chở khách, offline)
2. Tiêu chí chọn xe cho 1 request là xe ONLINE và có đường đi (google maps, fallback với harversine) đến vị trí khách là ngắn nhất.
3. Hệ thống chỉ tìm N lần cho 1 request (N có thể cấu hình trên server). Nếu đủ N lần mà không có xe nhận, request được ghi nhận là “KHÔNG CÓ XE”.
4. Có tất cả M điện thoại viên và N định vị viên làm việc đồng thời.
5. Hệ thống hoạt động realtime. Sinh viên tự cài đặt, có thể sử dụng các framework như **SocketIO**, **Pusher**, **OneSignal**. **KHÔNG ĐƯỢC PHÉP SỬ DỤNG** các cloud service như **FIREBASE**, …
6. Mọi api đều phải cài đặt JWT access-token và refresh-token